

UDI Network Drivers

Network Interface Metalanguage

<http://www.sco.com/forum1999/conference/developfast/F13>

Barry Feild

SCO Server Products Group

E-mail: barryf@sco.com



This is the eighth in a series of eight UDI presentations for SCO Forum 1999.

This session describes the UDI NIC Metalanguage and other aspects of UDI from the perspective of a Network Interface Card driver.

Agenda

- **SCO UDI Network Driver Architecture**
- **Network Interface Metalanguage Features**
- **Network Driver Initialization**
- **Network Driver Operations**
- **Functional Comparison of MDI and UDI**

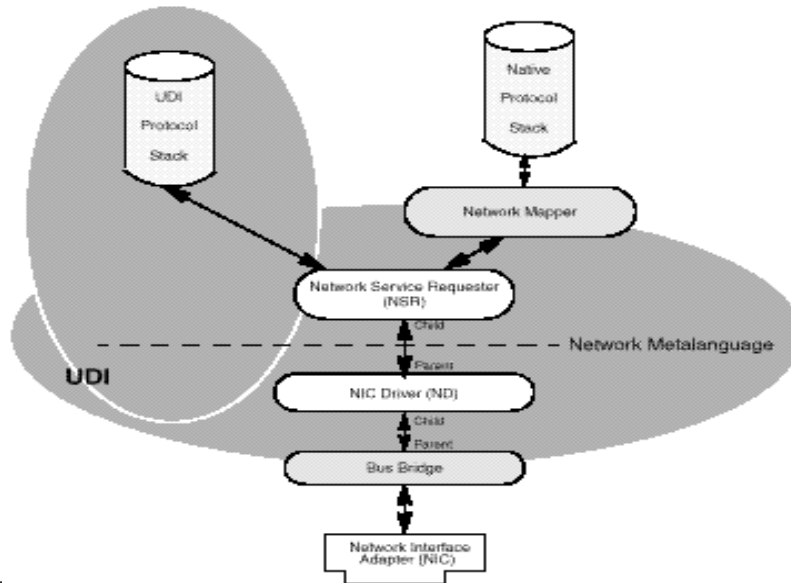


F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 2



UDI Network Driver Architecture

UDI Networking Environment

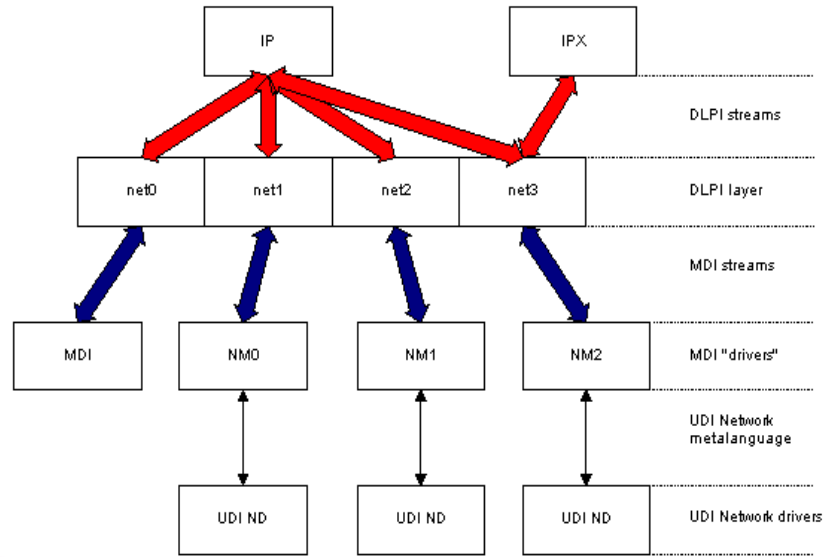


F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 3



SCO Network Driver Architecture

Initial Release

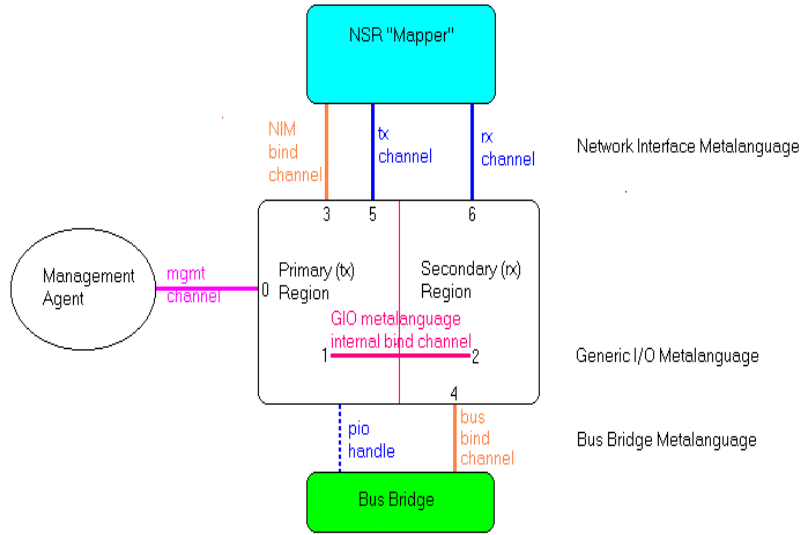


F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 4



SCO Network Driver Architecture

shrk UDI Driver



F13: UDI Network Drivers
 © 1999 SCO All Rights Reserved - Slide 5



Network Interface Metalanguage

UDI and Native Network Driver Comparison

- **UDI Advantages**

- Common programming model
- Portability
- Binary compatibility
- Implicit MP synchronization
- Simple ND interface

- **UDI Challenges**

- New technology



F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 6



UDI provides uniformity across device types; it defines a common execution model, inter-module communication and system services.

Network Interface Metalanguage

Network Driver/Network Service Requestor Interface

- **NIM: a universal set of connectionless network-related functions**
- **ND's primary task is send/receive data**
- **Single NSR-ND Interface active at any given time; separate channels are defined for:**
 - control/status
 - transmit data
 - receive data



F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 7



NIM was developed to provide all needed functionality in an OS, protocol stack, and transport independent manner.

NIM is designed to support a wide variety of network topologies

Network Interface Metalanguage

Network Service Requestor Responsibilities

- **Builds datalink headers for transmit packets**
- **Parse datalink headers on receive packets; NSR may perform SAP de-multiplexing**
- **Supports various MAC address lengths up to 20 bytes; not hard coded at a fixed size (such as 6 bytes)**



F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 8



Network Interface Metalanguage

NSR-ND Channels

- **Control Channel**
 - Bind/Unbind ND and NSR
 - Network Driver control operations (link state, MAC address registration, multicasting, statistics, etc.)
- **Receive Data Channel**
- **Transmit Data Channel**



F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 9



Network Interface Metalanguage

Flow Control

- **Transmit Flow Control**

- Tx control blocks owned by ND, loaned to NSR
- ND returns control blocks to NSR on tx complete
- NSR never allocates transmit blocks

- **Receive Flow Control**

- Rx control blocks owned by NSR, filled by ND
- ND never allocates receive buffers
- Rx buffers may be recycled by NSR



F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 10



Tx control block includes a completion urgency hint to indicate to the ND the importance of returning the tx buffer/control block quickly.

`udi_net_tx_cb_t completion_urgent` is typically set when the buffer containing the packet may be a critical resource, e.g., it's been loaned to the networking subsystem from another OS module such as NFS/esballoc.

`Completion_urgent` may also be set when the NSR thinks it's running out of tx buffers

Receive flow control is essentially the same as transmit flow control, but it's been implemented in the reverse direction.

Network Interface Metalanguage

Hardware Checksum Offloads

- **Buffer tags used to support CRC offload**
 - `udi_buf_tag_compute()`
 - `udi_buf_tag_apply()`
- **Transmit checksum capability handled separately from Receive checksum capability**
- **Buffer tags may be used for arbitrary per-buffer data**



F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 11



Network Driver Initialization

Static Driver Properties

- **Static Driver Properties**

- Region Declarations
- Parent/Child/Internal bind_ops Declarations
- Custom Declarations
 - » Adapter-specific configuration parameters
 - » ND reads these attributes on link-enable requests using `udi_instance_attr_get()`



F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 12



Network Driver Initialization

Network Driver Channel Ops Vector Registration

- **Control channel ops - udi_nd_ctrl_ops_t**
 - channel events
 - bind/unbind/enable/disable
 - control requests (MAC addrs, multicasting, stats)
- **Transmit channel ops - udi_nd_tx_ops_t**
 - normal/expedited transmit requests
- **Receive channel ops - udi_nd_rx_ops_t**
 - NSR gives receive control blocks/buffers to ND.



F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 13



Network Driver Initialization

UDI Control Block Registration

- **Control blocks (and Ops vectors) registered in udi_init_info structures:**
 - udi_cb_init_t
 - udi_ops_init_t
- **All channel operations are done with appropriate ctrl or transfer control block**
- **All types of control blocks may be allocated internally with udi_cb_alloc()**



F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 14



Network Driver Initialization

Network Driver Instantiation

- **Parent driver (Bus Bridge) instantiated, mgmt channel established; MA sends enumeration request to parent**
- **MA creates ND instance (primary and secondary regions, mgmt channel, internal bind channels) that corresponds to parent enumeration response info**
- **MA issues udi_usage_ind() on mgmt channel - first operation on newly instantiated driver instance**



F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 15



Network Driver Initialization

Network Driver Instantiation (continued)

- **MA begins child/parent bind sequence; issues UDI_CHANNEL_BOUND on child (ND) end of ND-Bus bind channel**
 - may be in primary region, static or dynamic secondary region
- **ND performs internal initialization, reads instance attributes, binds to parent with `udi_bus_bind_req()`**



F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 16



Network Driver Initialization

Network Driver Instantiation (continued)

- **Parent (Bus Bridge) processes ND's bind request, issues `udi_bus_bind_ack()`**
- **ND completes initialization, issues `udi_channel_event_complete()` for the parent bind operation to the MA**
- **Process repeated for ND (parent) and NSR (child) using NIM bind operations**



F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 17



Network Driver Initialization

Network Driver Enumeration Attributes

- **Network Driver Enumeration Attributes**

- if_num: port instance number (32bit unsigned)
- if_media: media type (macro)
- identifier: media type (string)
- address_locator: port instance number (string)
- physical_locator: interface MAC address (string)



F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 18



Network Driver Initialization

Network Interface Metalanguage Bind Operation

- **MA creates NSR-ND initial bind channel (control channel) using Network Interface Metalanguage ctrl_ops role**
 - Network Interface Metalanguage control channel is now established
- **NSR issues ND udi_nd_bind_req() on NSR bind channel**
 - ND propagates constraints (memory requirements for DMA, etc.) to NSR with udi_constraints_propagate()



F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 19



Network Driver Initialization

Network Interface Metalanguage Bind Operation (cont)

- **ND actions on `udi_nd_bind_req()` (cont)**
 - ND spawns and anchors transmit channel, using transmit channel ops vector index
 - ND spawns and anchors receive channel, using receive channel ops vector index
- **ND acks bind with `udi_nsr_bind_ack()`**

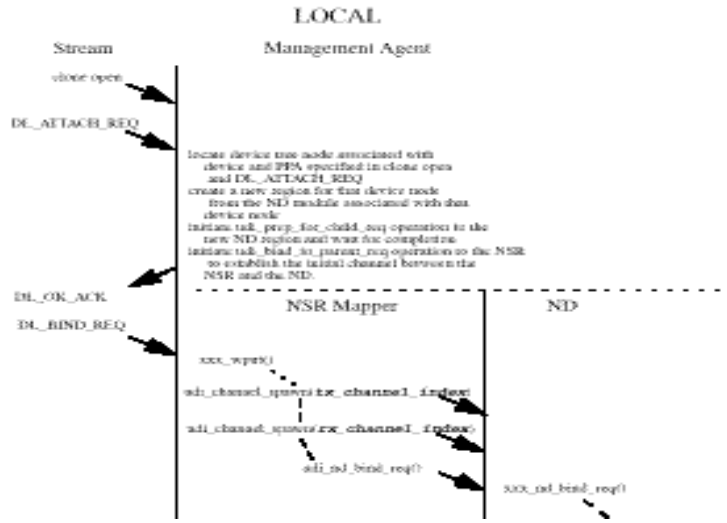


F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 20



Network Driver Initialization

Connectionless Network Bind Operation

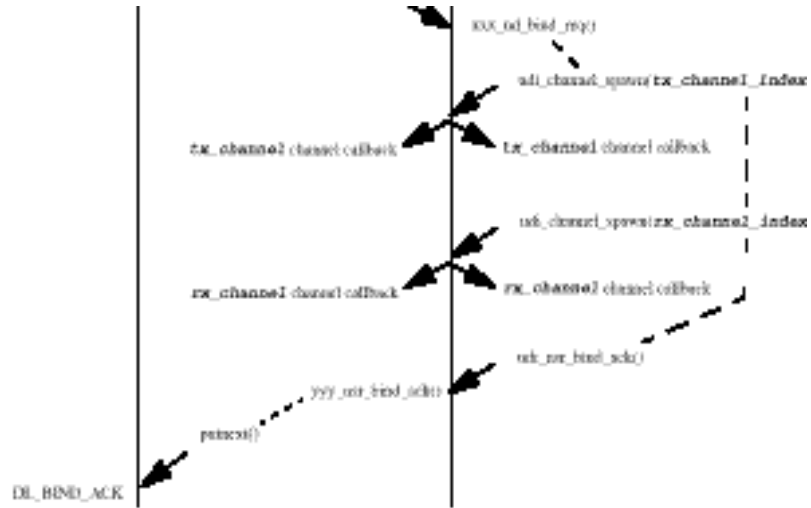


F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 21



Network Driver Initialization

Connectionless Network Bind Operation (continued)



F13: UDI Network Drivers
 © 1999 SCO All Rights Reserved - Slide 22



Network Driver Initialization

Network Interface Metalanguage Unbind Operation

- **Unbind used to detach channels and release resources established during Network Bind operation**
- **ND acks with `udi_nsr_unbind_ack()`**
- **ND closes control and both data channels with `udi_channel_close()`**



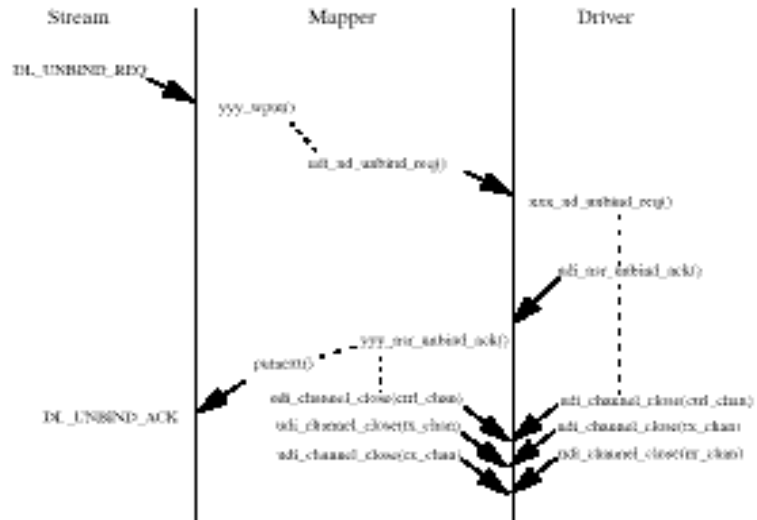
F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 23



Network Driver Initialization

Connectionless Network Unbind Operation

LOCAL



F13: UDI Network Drivers
 © 1999 SCO All Rights Reserved - Slide 24



Network Driver Operations

Network Interface Control

- **Enable network interface link activity**
 - udi_nd_enable_req()
 - udi_nsr_enable_ack()
- **Disable network interface link activity**
 - udi_nd_disable_req() - no ack
- **Indicate link state change**
 - udi_nsr_status_ind()



F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 25



Network Driver Operations

Control and Status Operations

- **Control Requests - udi_nd_ctrl_req()**
 - UDI_NET_ADD_MULTI
 - UDI_NET_DEL_MULTI
 - UDI_NET_ALLMULTI_ON
 - UDI_NET_ALLMULTI_OFF
 - UDI_NET_GET_CURR_MAC
 - UDI_NET_SET_CURR_MAC
 - UDI_NET_GET_FACT_MAC



F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 26



Network Driver Operations

Control and Status Operations (continued)

- **udi_nd_ctrl_req() (continued)**
 - UDI_NET_PROMISC_ON
 - UDI_NET_PROMISC_OFF
 - UDI_NET_HW_RESET
 - UDI_NET_BAD_RXPKT
- **Control Request ack - udi_nsr_ctrl_ack()**



F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 27



BAD_RXPKT specifies ND handling of rx packets with errors.
Sent from NSR to ND.

Network Driver Operations

Control and Status Operations (continued)

- **Status Indications - udi_nsr_status_ind()**

- UDI_NET_LINK_UP
- UDI_NET_LINK_DOWN
- UDI_NET_LINK_RESET

- **Statistics**

- udi_nd_info_req()
- udi_nsr_info_ack()
- udi_net_info_cb_t



F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 28



Network Driver Operations

Control and Status Operations (continued)

- **Statistics (cont) - udi_net_info_cb_t**

- interface_is_active
- link_is_active
- is_full_duplex
- link_mbps
- link_bps
- tx_packets
- rx_packets
- tx_errors
- rx_errors
- tx_discards
- rx_discards
- tx_underrun
- rx_overrun
- collisions



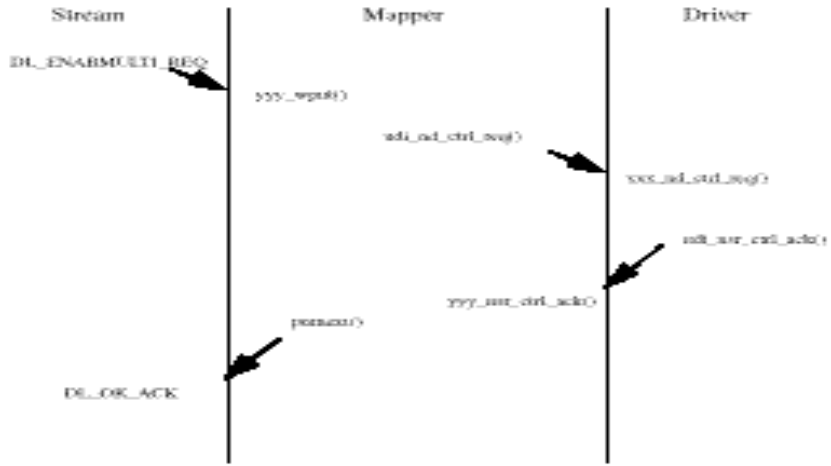
F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 29



Network Driver Operations

Control Operation (enable multicast address)

LOCAL



F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 30



Network Driver Operations

Data Transfer - Transmit

- **ND owns transmit buffers/control blocks**
 - udi_cb_alloc()
- **ND loans transmit control blocks to NSR**
 - udi_nsr_tx_rdy()
- **NSR sends transmit packets to ND**
 - udi_nd_tx_req()
 - udi_nd_exp_tx_req()



F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 31



Network Driver Operations

Data Transfer - Receive

- **NSR owns receive control blocks**
 - udi_cb_alloc()
- **NSR loans receive control blocks to ND**
 - udi_nd_rx_rdy()
- **ND sends receive packets to NSR**
 - udi_nsr_rx_ind()
 - udi_nsr_exp_rx_ind()

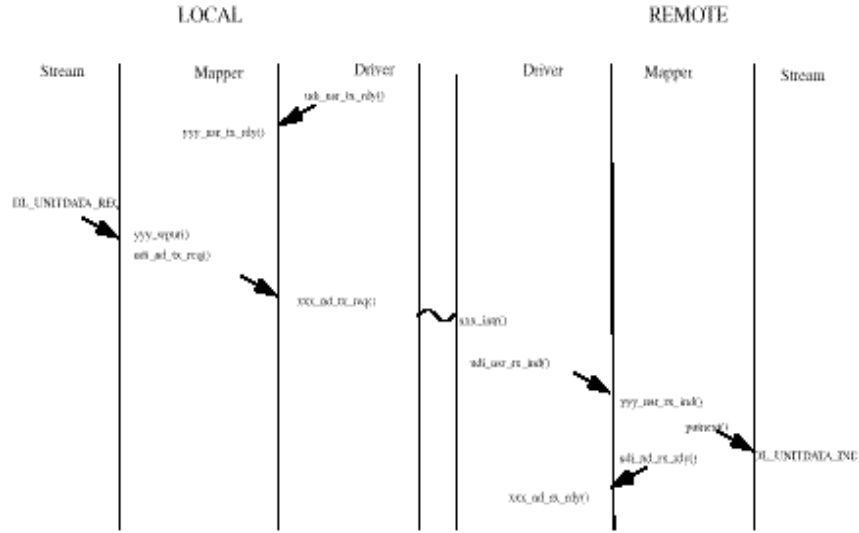


F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 32



Network Driver Operations

Connectionless Data Transfer Operations



F13: UDI Network Drivers
 © 1999 SCO All Rights Reserved - Slide 33



MDI/UDI Functional Comparison

MDI Primitives

- **MDI Primitives**

- MAC_BIND_REQ
- MAC_OK_ACK
- MAC_INFO_REQ
- MAC_INFO_ACK
- MAC_ERROR_ACK
- MAC_HWFAIL_IND
- MAC_HWSUSPEND_IND
- MAC_HWRESUME_IND

- **UDI Equivalent**

- udi_nd_bind_req()
- udi_nd_bind_ack()
- udi_nd_info_req()
- udi_nsr_info_rsp()
- udi_nsr_bind_res error
- UDI_NET_HW_RESET
- UDI_DMGMT_SUSPEND
- UDI_DMGMT_RESUME



F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 34



MDI/UDI Functional Comparison

MDI ioctl

- **MDI ioctl**

- MACIOC_SETALLMCA
- MACIOC_DELALLMCA
- MACIOC_SETMCA
- MACIOC_DELMCA
- MACIOC_GETADDR
- MACIOC_GETRADDR
- MACIOC_SETADDR
- MACIOC_GETSTAT
- MACIOC_PROMISC

- **UDI Equivalent ctrl_ops**

- UDI_NET_ALLMULTI_ON
- UDI_NET_ALLMULTI_OFF
- UDI_NET_ADD_MULTI
- UDI_NET_DEL_MULTI
- UDI_NET_GET_CURR_MAC
- UDI_NET_GET_FACT_MAC
- UDI_NET_SET_CURR_MAC
- udi_nd_info_req()
- UDI_NET_PROMISC_ON/OFF



F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 35



MDI/UDI Functional Comparison

MDI ioctls (continued)

- **MDI ioctls with no UDI equivalent**
 - MACIOC_SETSTAT
 - MACIOC_CLRSTAT



F13: UDI Network Drivers
© 1999 SCO All Rights Reserved - Slide 36



F13: UDI Network Drivers

15, August, 1999

